Cognitive Delve on Cryptography- Proffer of New Algorithm

D Paul Joseph[1], K Arun[2], V Srinivasa Rao[3]
[1] *Dept. of Computer Science and Technology*
Sir CRR College of Engineering, Eluru
[2] *Asst. Professor, Dept. of Computer Science and Engineering*
SSN College of Engineering, Ongole
[3] *Asst.Professor, Dept. of Computer Science and Engineering*
Narasaraopeta Institute of Technology, Narasaraopeta
[1]Pauljoseph91@gmail.com
[2]Karun014@gmail.com
[3]vallepucnu@gmail.com

***Abstract*-- The present era resembles the digital world, in which everything irrespective of technology, information is digitalized. The task of digital world is to present the data across the global without any interruptancy, intruder attacks or exploit attacks. To achieve this, all the sensitive information that is to be crawl over the earth's surface is changed into cipher text which is unreadable or unknown to other sources. For this encryption, we go for different algorithms like symmetric and asymmetric such as DES, 3DES, AES, RSA, Blowfish, Hashing algorithms and so on. But in fact each and every algorithm has some cons in matter of size, speed, time and breakage of cipher to some extent. So this paper proposes a new hybrid algorithm which works fine in all cases and provides a fair comparison with so called well developed encryption and decryption algorithms. The proposed algorithm works in very less time and is proved to be efficient in case of time and breakage of cipher.**

I. INTRODUCTION

Cryptography is the art of writing information in unreadable format or unpredictable format. The word Cryptography is derived from the Greek language compromising two words kryptós meaning hidden, and gráphein meaning to write - or hidden writing. Cryptography is the study of information hiding and verification. It includes the protocols, algorithms and strategies to securely and consistently prevent or delay unauthorized access to sensitive information and enable verifiability of every component in a communication. People who study and develop cryptography are called cryptographers. The study of how to circumvent the use of cryptography for unintended recipients is called cryptanalysis, or code breaking. Cryptography and cryptanalysis are sometimes grouped together under the same term called cryptology.

Cryptology deals with the algorithms which can cipher and decipher the text i.e.., encrypt and decrypt the information whether the information is text, multimedia information or predicates. Some of the algorithms so far used are classified into two categories such as symmetric and asymmetric algorithms or can also be called as private and public key algorithms.

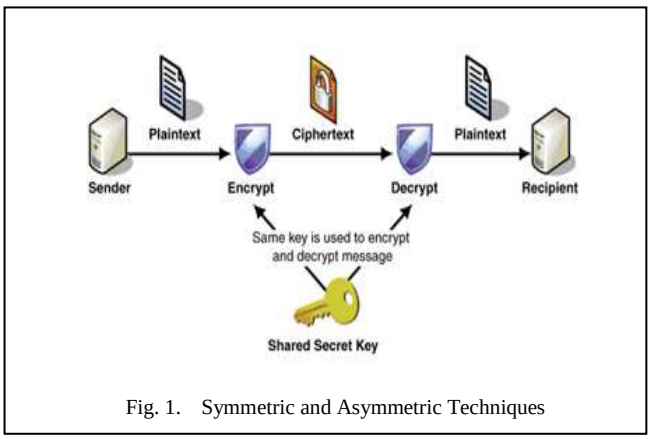

Fig. 1. Symmetric and Asymmetric Techniques

Symmetric algorithms uses only one key which is termed as private key and this key must be known to both sources. These mainly work efficiently on hardware platform rather than software platforms. Asymmetric key algorithm uses two keys, which can be defined as private key and public key. Private Key is used for encryption purpose whereas public key is used for decryption purposes. These algorithms are also called as public key algorithms. In advantage to private key, public key algorithms uses computational and complex mathematical methods.

II. GOALS AND TERMS OF CRYPTOGRAPHY

The common and most essential goals of cryptography can be classified into four categories. They are described as follows.

i. Confidentiality or Privacy
ii. Integrity
iii. Authentication and
iv. Non-Repudiation

Confidentiality or Privacy ensures that only an authorized recipient should be able to extract the contents of the message from its encrypted form. Message integrity ensures that the recipient should be able to determine if the message has been altered or not. Authentication clears that the recipient should be able to verify from the message, the identity of the sender, the origin or the path it traveled (or combinations) so to validate claims from emitter or to validated the recipient expectations. Non-repudiation states that the emitter should not be able to deny sending the message. There are, in general, three types of cryptographic schemes typically used to accomplish these

goals: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions.

Coming to Terminology of cryptology, most often used terms are:

i. Symmetric key or Private key
ii. Asymmetric key or Public key
iii. Encryption and
iv. Decryption.

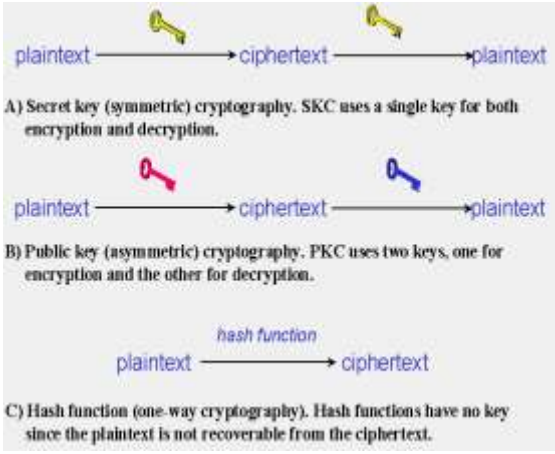

Fig. 2. Types of Cryptography Algorithms

III. SYMMETRIC ALGORITHMS

In symmetric key algorithms, the same key is used for both encryption and decryption. Simply it can be understood as both the sender and the receiver uses same key to send or receive the message. Typically there are few algorithms which fall under this category. These can be classified as:

- Block Ciphers
- Stream Ciphers
- DES
- 3DES
- AES
- Skipjack.
- Cast 128/256
- International Data Encryption Algorithm
- Rivest Ciphers
- Blowfish
- Kcipher and so on.

A. DATA ENCRYPTION STANDARDS ALGORITHM

The most common used algorithm- DES was designed by IBM in the 1970s and adopted by the National Bureau of Standards. DES is a block-cipher employing a 56-bit key that operates on 64-bit blocks. DES has a complex set of rules and transformations that were designed specifically to yield fast hardware implementations, slow software implementations.

DES, 16 cycle Feistel system is used, with an overall 56-bit key permuted into 16 48-bit sub keys, one for each cycle. For decryption, the same algorithm is used, with the order of sub keys reversed. The Left and Right blocks are 32 bits each (4bytes), totaling an overall block size of 64 bits. The hash function "f" uses "S-boxes", which takes a 4-byte data block and one of the 6-byte sub keys as input and produces 4bytes of output.

B. TRIPLE DATA ENCRYPTION STANDARD ALGORITHM

3DES is the enhanced version of the DES algorithm. Since the DES seemed to be less efficient because of dictionary and brute force attacks, design of new algorithm was essential. In 3DES, it follows three steps.

For encryption it follows:

Encryption – Decryption - Encryption

For decryption it follows:

Decryption – Encryption – Decryption.

This standard specifies three keying options:

- Keying option 1: All three keys are independent
- Keying option 2: K_1 and K_2 are independent, and $K_3 = K_1$
- Keying option 3: All three keys are identical, i.e. $K_1 = K_2 = K_3$
- Keying option 1: the key space is 56 x 3= 168 bits
- Keying option 2 provides less security than option 1, with $2 \times 56 = 112$ key bits
- Keying option 3 is equivalent to DES, with only 56 key bits. This option provides backward compatibility with DES

Drawbacks:

- Encrypt: C = EK3 [DK2 [EK1 [P]]]
- Decrypt: P = DK1 [EK2 [DK3 [C]]]

If we use three completely different keys, will there be 168bits effectively strength?

C. *SKIPJACK*

Skipjack uses an 80-bit key to encrypt or decrypt 64-bit data blocks. It is an unbalanced Feistel network with 32 rounds. It was designed to be used in secured phones. Skipjack was proposed as the encryption algorithm in a US government-sponsored scheme of key escrow, and the cipher was provided for use in the Clipper chip, implemented in tamper proof hardware. Skipjack is used only for encryption; the key escrow is achieved through the use of a separate mechanism known as the Law Enforcement Access Field (LEAF).

Attacks: Impossible differential cryptanalysis

A truncated differential attack was also published against 28 rounds of Skipjack cipher.

A claimed attack against the full cipher was published in 2002.

D. *CAST 128/256*

It is a DES-like substitution-permutation crypto algorithm, employing a 128-bit key operating on a 64-bit block.CAST-256 (RFC 2612) is an extension of CAST-128, using a 128-bit block size and a variable length (128, 160, 192, 224, or 256 bit) key. CAST is named for its developers, Carlisle Adams and Stafford Tavares and is available internationally. CAST-256 was one of the Round 1 algorithms in the AES process.

E. *International Data Encryption Algorithm*

The International Data Encryption Algorithm (IDEA), originally called Improved Proposed Encryption Standard (IPES), is a symmetric-key block cipher designed by James Massey of ETH Zurich and Xuejia Lai and was first described in 1991. The algorithm was intended as a replacement for the Data Encryption Standard (DES). IDEA is a minor revision of an earlier cipher, Proposed Encryption Standard (PES). IDEA was used in Pretty Good Privacy (PGP), and was incorporated after the original cipher used in v1.0, IDEA is an optional algorithm in the Open PGP standard.

Many Weak keys are identified in this algorithm.

F. *Rivest Ciphers*

Rivest Ciphers named for Ron Rivest, a series of SKC algorithms.

RC1: Designed on paper but never implemented, and so it was not in existence.

RC2: A 64-bit block cipher using variable-sized keys designed to replace DES.

RC3: Found to be breakable during development, and so the future enhancement remained hurdle.

RC4: A stream cipher using variable-sized keys; it is widely used in commercial cryptography products. An update to RC4, called Spritz, was designed by Rivest and Jacob Schuldt

RC5: A block-cipher supporting a variety of block sizes (32, 64, or 128 bits), key sizes, and number of encryption passes over the data.

RC6: A 128-bit block cipher based upon, and an improvement over, RC5; RC6 was one of the AES Round 2 algorithms.

G. *Blowfish Algorithm*

A symmetric 64-bit block cipher invented by Bruce Schneier; optimized for 32-bit processors with large data caches, This is significantly faster than DES on the PowerPC-class machine. Key lengths can vary from 32 to 448 bits in length. Blowfish, available freely and intended as a substitute for DES or IDEA, is in use in a large number of products. Blowfish is a fast block cipher, except when changing keys. Each new key requires pre-processing equivalent to encrypting about 4 kilobytes of text, which is very slow compared to other block ciphers. This prevents its use in certain applications. Blowfish is known to be susceptible to attacks on reflectively weak keys. This means Blowfish users must carefully select keys as there is a class of keys known to be weak, and Blowfish should not be used to encrypt files that are larger than 4GB because of its small 64-bit block size.

H. *Advanced Encryption Standard Algorithm*

The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits. The algorithm specified in this standard may be implemented in software, firmware, hardware, or any combination thereof. The specific implementation may depend on several factors such as the application, the environment, the technology used, etc. Block Size: 128bits

Consists of Four Operations:

1) Substitute Bytes

Sub byte is only nonlinear which substitutes all bytes of the state array using a LUT which is a 16x16 matrix of bytes, often called S-box Units.

2) Shift Rows

The main goal of this process is to correlate and scramble the byte order inside each 128-bit block. In the shift the bytes in the last three rows of the State are cyclically shifted over different numbers of bytes (offsets).

3) Mix Column Transformation

This transformation is based on Galois Field multiplication. Each byte of a column is replaced with another value that is a function of all four bytes in the given column. The Mix

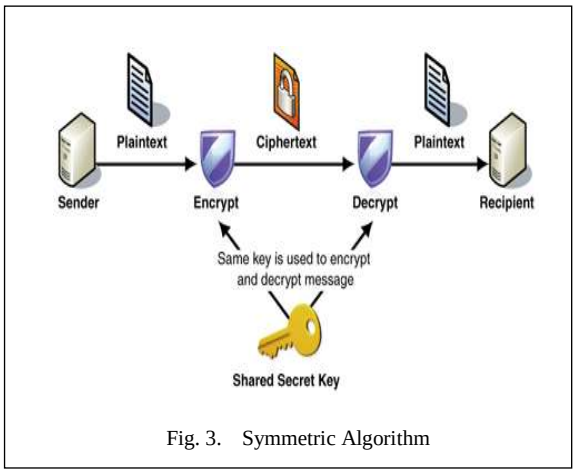

Fig. 3. Symmetric Algorithm

Columns transformation is performed on the State column-by column.

4) Add Round Key

In this operation, the round key is applied to the State by simple bit by bit XOR. Key Expansion unit generates the next round key as for three different key sizes.

IV. Disadvantages of Symmetric Techniques

A major problem with such a system is that the sender and receiver must know the key prior to transmission. This requirement makes such a system difficult to use in practice. The key cannot be openly transmitted since that would compromise the security of system. One possibility is for the two parties to meet and exchange the keys prior to transmitting their messages. However, this exchange becomes more difficult when many parties are involved in a communications network.

Secondly, most of the symmetric algorithms contain more number of rounds, which leads to large processing time. Thirdly, these algorithms contain a bit of smaller block sizes, which may lead to numerous times of block division. Since these contains only single key for all transmissions, if that key is anyhow known, then whole transmission would be a failure.

Another disadvantage is that there is no provision for data origin authentication and data integrity protection. In other words, the recipient can neither authenticate the sender nor verify that the decrypted message is the same as the original message and finally cannot provide digital signatures that cannot be repudiated.

V. ASYMMETRIC ALGORITHMS

Asymmetric cryptography algorithms use two keys which can be referred as public key and private key. Public key is used for encryption purpose and private key is used for decryption purpose. These two keys are mathematically related, but it is very difficult to obtain one from the other unless one knows the transformation. The public key can be revealed without compromising the security of the system. The corresponding private key, however, must not be revealed to any party. The different algorithms designed under this category are:

a. RSA
b. Diffie-Hellman
c. Digital Signature
d. ElGamal
e. Elliptic Curve
f. XTR

a) Rivest, Adi shamir, Leonard Adleman Algorithm

Rivest-Shamir-Adleman is the most commonly used asymmetric algorithm (public key algorithm). It can be used both for encryption and for digital signatures. The security of RSA is generally considered equivalent to factoring, although this has not been proved. The key size should be greater than 1024 bits for a reasonable level of security. RC5 is one of the most common in use. RSA encryption is a deterministic encryption algorithm (i.e., has no random component). An attacker can successfully launch a chosen plaintext attack against the cryptosystem, by encrypting likely plaintexts under the public key and test if they are equal to the cipher text. Secondly it has the property that the product of two cipher texts is equal to the encryption of the product of the respective plaintexts. That is $m_1^e m_2^e \equiv (m_1 m_2)^e \pmod{n}$, which results in chosen-cipher text attack is possible.

b) Diffie Hellman

Diffie-Hellman is the first asymmetric encryption algorithm, invented in 1976, using discrete logarithms in a finite field. Allows two users to exchange a secret key over an insecure medium without any prior secrets. The key exchange by Diffie-Hellman protocol remedies this situation by allowing the construction of a common secret key over an insecure communication channel. It is based on a problem related to discrete logarithms, namely the Diffie-Hellman problem. This problem is considered hard, and it is in some instances as hard as the discrete logarithm problem. The Diffie-Hellman protocol is generally considered to be secure when an appropriate mathematical group is used. Usually, Diffie-Hellman is not implemented on hardware.

c) Digital Signature

Digital Signature Algorithm (DSA) is similar to the one used by ElGamal signature algorithm. It is fairly efficient though not as efficient as RSA for signature verification. The standard defines DSS to use the SHA-1 hash function exclusively to compute the message digests. This algorithm is also called as Message Digest Algorithm. The main problem with DSA is the fixed subgroup, which limits the security to around only 80 bits. Hardware attacks can be menacing to some implementations of DSS. However, it is widely used and accepted as a good algorithm.

d) ElGamal

The ElGamal is a public key cipher - an asymmetric key encryption algorithm for public-key cryptography which is based on the Diffie-Hellman key agreement. ElGamal is the predecessor of DSA. ElGamal encryption consists of three components: the key generator, the encryption algorithm, and the decryption algorithm. ElGamal encryption is an unconditionally malleable, and therefore is not secure under the chosen cipher-text-attack. Crypto systems like The Cramer–Shoup cryptosystem and DHAES are proved to be deploit this algorithm.

e) Elliptic Curve

Elliptic Curve DSA (ECDSA) is a variant of the Digital Signature Algorithm (DSA) which operates on elliptic curve groups. As with Elliptic Curve Cryptography in general, the bit size of the public key believed to be needed for ECDSA is about twice the size of the security level, in bits. Timing attack using Open SSL can be used to break the security of this algorithm.

Secure Random Collisions also result in breakage of this.

f) XTR

XTR stands for 'ECSTR', which is an abbreviation for Efficient and Compact Subgroup Trace Representation. It is a method to represent elements of a subgroup of a multiplicative group of a finite field. XTR relies on the difficulty of solving Discrete Logarithm related problems in the full multiplicative group of a finite field. It is a novel method that makes use of traces to represent and calculate powers of elements of a subgroup of finite. It is based on the first public key cryptosystem protocol known as the Diffie-Hellman key agreement protocol. Advantages of XTR are its fast key generation, small key sizes, and speed.

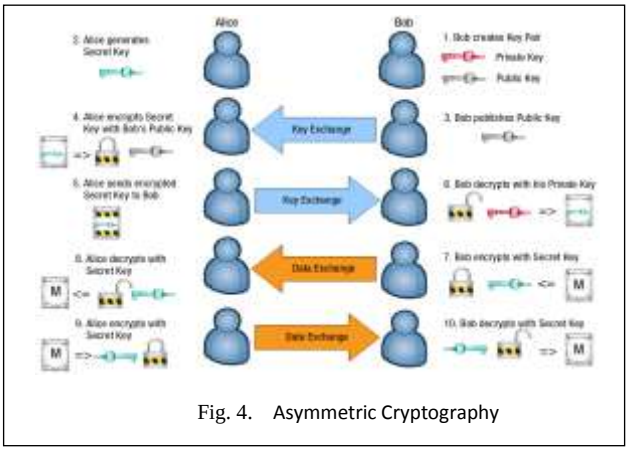

Fig. 4. Asymmetric Cryptography

VI. DISADVANTAGES OF ASYMMETRIC TECHNIQUES

Public key algorithms are theoretically easier to attack than symmetric key algorithms because the attacker (presumably) has a copy of the public key that was used to encrypt the message. The job of the attacker is further simplified because the message presumably identifies which public key encryption algorithm was used to encrypt the message. Public key algorithm attacks generally fall into two categories

i. Key search attacks
ii. Analytic attacks.

Asymmetric algorithms uses larger key sizes which results in the increase of execution time. Secondly these uses complexual

mathematical methods, which are difficult to implement and time taking to execute.

I. COMPARISON OF SYMMETRIC AND ASYMMETRIC CRYPTOGRAPHY ALGORITHMS

- Larger the block size→slower execution
- Larger key size→High security
- Increased Rounds→more execution time
- Larger key size→more arithmetical operations →More consumption of time.

The in detail comparison of some known public and private key algorithms are given in the following table.

Table I. SYMMETRIC VERSUS ASYMMETRIC

Algorithms	***IDEA***	***DES***	***3DES***	***AES***	***RSA***	***MD5***	***Elliptical Curve***
Author	James Masey, uejia Lai	IBM	IBM	Joan Daemen, Incent	Rivest,Shamir, dleman	Ronald Rivest	Neal Kobiltz, Victor S Miller
Year	1991	1975	1978	1998	1977	1992	1985
Structure	Symmetric Block cipher	Festial	Festial	Substitution Permutation	Public key algorithm	Merkle mgard	Algebric structure of
Rounds	8.5	16	48	10, 12, 14	1	4	Based on Point multiplication
Key (bits)	128	56	168	128, 192, 256	Greater than 1024bits	512	160-521
Block size (in bits)	64	64	64	128	128	512	Less than that of RSA
Security	Inadequate Strong	Vulnerable	Adequate vulnerable	Strongly Ciphered	High security	Moderately secured	High security and strong enough
Execution speed	Moderate	Moderate	Moderate	Faster	Slower	Moderate	Moderate
Vulnerabilities	Weak-keys, Differential Crypt analysis	Brute-Force, Cryptanalysis	Cryptanalysis	Brute force(not yet proved)	Oracle attacks	Collision, Preimage vulnerability	Side channel, Quantam computing
Power Consum	Low	Low	Low	Low	High	Moderate	Moderate
Encryption/Decryption speed	Faster	Moderate	Moderate	Faster	Low	Faster	Faster
Possible Keys	2^128	2^56	2^112, 2^168	2^128, 2^192, 2^256	2^128	2^512	>2^512
Resultant size	Resultant text size is lesser or equal to original text.				Resultant text size is greater than original text.		
Type of Alg	Symmetric Algorithm				Asymmetric Algorithm		
Usage	Can be used for only Encryption and the decryption (Confidentiality).				Can be used for Confidentiality as well as Integrity and Non-repudiation checks.		

VII. PROPOSED ALGORITHM

Based on the different cons of various said algorithms, in this paper a new hybrid algorithm was proposed by inheriting the properties of AES and block cipher algorithms. First take the key of size 256bits≡32Bytes. Divide it into two blocks of matrices of size 4X4 such that each matrix is of 16bytes. Now perform XOR operation between two matrices K1 and K2 such that XORing is based on the numbering and position of indices and store the resultant matrix in K1. Since the plain text can be of any size, divide the plain text into blocks of size 128bytes≡16Bytes, and number/name each block. Arrange each block into 4x4 matrix such that each cell is of size 1Byte (numbering starts from 0-15). Perform XOR operation on each block such that 1st cell is XOR'ed with last cell and 2nd cell with 14th cell, except for the case 8th cell is XOR'ed with 8th cell itself. The resultant matrix consists of self XORed text. Now the obtained matrix is again XOR'ed with Key K1 and stored in the each block itself and repeat for other blocks.

Now take another key K3 of size 16Bytes and XOR with the obtained matrix consisting of cipher text, such that more security is provided and any hacker or intruder finds difficult to know the original text. Now combine the result of every block/matrix and send to the opposite source. The cipher text is undergone through 3XOR operations by providing 2keys. Now repeat the above process in reverse to get the original message.

Proposed Algorithm:

1. Select two keys public key K1 and private key K2.
2. K1=256bits≡32bytes and K2=128bits≡16bytes.
3. Divide K1 into two 4x4 matrices such that each matrix is of size 16bytes and each cell is of 1byte.
4. Now perform XOR operation on two 4x4 matrices and store in the matrix K1.
5. Divide the plain text into blocks size of 128bits≡16bytes, until the plain text reaches end.
6. Now arrange each block of plain text into 4x4 matrix and name it as B1, B2, B3,……..Bn.
7. Perform XOR operation on each block such that the every 1st cell is XOR'ed with 15th cell and 2nd cell with 14th cell, except for the case 8th cell is XOR'ed with 8th cell itself., repeat the same process for all blocks and store the result in the same blocks.
8. Now perform the Xor operation on every block with key K1 matrix and store the result in the same blocks.
9. Take private key K2 of size 16bytes and again perform the XOR operation with each and every block.
10. Combine all the data from each matrix and send to other source.
11. Since the reverse operation for XOR is XOR itself, if the whole process is repeated in reverse order, we get the original text.

VIII. ADVANTAGES OVER THE PRIOR ALGORITHMS

1. Since it just uses only XOR operations rather than complex mathematical functions, execution speed can be faster.
2. Though many XOR operations are to be performed, as XOR is simple to perform, it works efficiently.
3. As it uses 256bit key size, it will take 2^256 combinations and it is really hard to break the cipher.
4. As it consists of two keys, it is very difficult to break the text.
5. If one key is known to the attacker without knowledge of second key, it is hardly impossible to break cipher.
6. Since the block size is of 128bit(less when compared to other ones), the execution will be faster.
7. No need of extra buffers and temporary variables so that it saves more memory and space.
8. As this algorithm uses just simpler XOR operations, the time complexity will be <O (log n) and in worst case it may be O (n log n), where n is number of rounds.

IX. CONCLUSION

In this paper a new comparative study between symmetric and asymmetric algorithms were presented on various factors which are key length, cipher type, block size, developed, possible keys, and new algorithm was proposed for encryption and decryption of information. Our theoretical analysis resulted that when compared to the prior algorithms, it works efficiently in the case of execution time, breakage of cipher, efficiency and security. Still this is in practical stage, we hope this algorithm would yield expected result and withstands to exploits.

Our future work will focus on implementation and simulation of this algorithm and continue to work on multimedia data.

REFERENCES

[1] [Bruce1996] BRUCE SCHNEIER, "Applied Cryptography", John Wiley & Sons, Inc. 1996

[2] Dawn Xiaodong Song David Wagner Adrian Perring "Practical Techniques for Searches on Encrypted Data" Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on 2002.

[3] D Paul Joseph, K Arun, M Krishna "Cognitive Analytics and Comparison of Symmetric and Asymmetric Cryptography Algorithms" in International Journal of Advanced Research in Computer Science and Software Engineering in Volume6, May 2015.

[4] [Nadeem2005] Aamer Nadeem et al, "A Performance Comparison of Data Encryption Algorithms", IEEE 2005

[5] Abdel-Karim Al Tamimi, Swati. "Performance Analysis of Data Encryption Algorithms ", International Journal of Advanced Research in Computer Science and Software Engineering 3(2), pp. 147-149, February – 2013.

[6] Nidhi Singhal1, J.P.S.Raina2," Comparative Analysis of AES and RC4 Algorithms for Better Utilization", International Journal of Computer Trends and Technologypp.177-181, Aug 2011.

[7] Pratap Chandra Mandal, " Evaluation of performance of the Symmetric Key Algorithms: DES, 3DES ,AES and Blowfish ", Journal of Global Research in Computer Science Department of Computer Application, vol 3, pp. 67-70, August 2012.

Selected Paper from International Conference on Computing (NECICC-2k15)